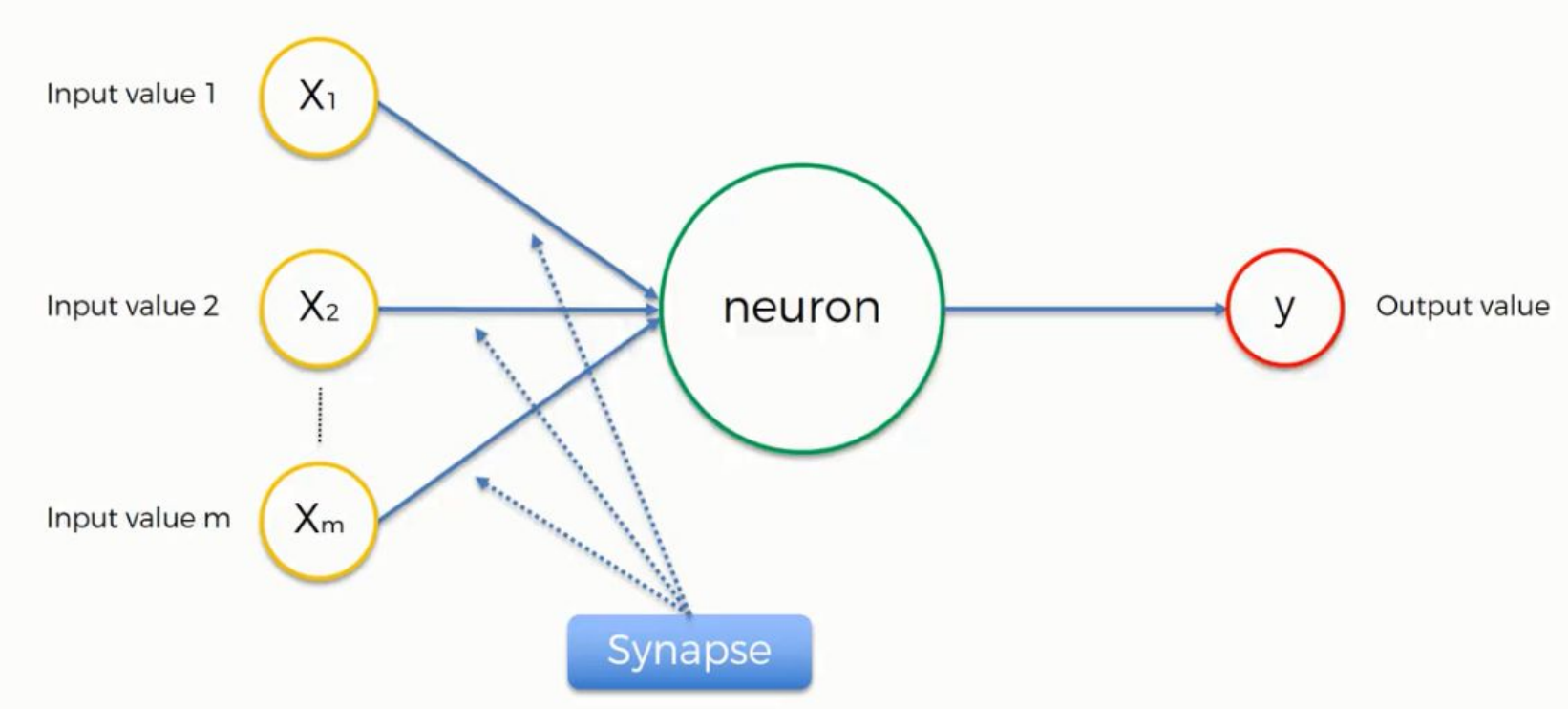
Artificial Neural Network

Introduction

- A simple machine takes the input and converts it into an output.The same input will always provide same output.
- But when we consider human brain, has a unique characteristic of creating transient states through neurons in between the sensory organs and the brain.
- These transient states create randomness in decision making.
- So, in machine learning algorithms or artificial neural networks, transient states which allow machines to learn in a more sophisticated manner.

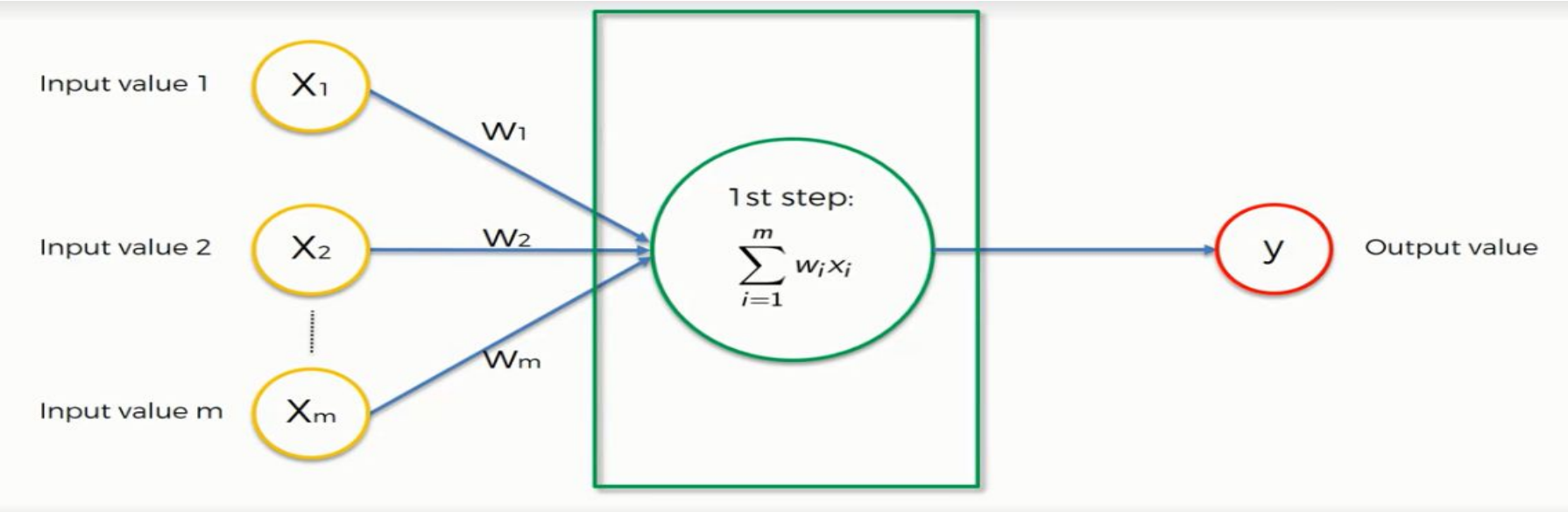
Basic structure of ANN



Basic structure of ANN

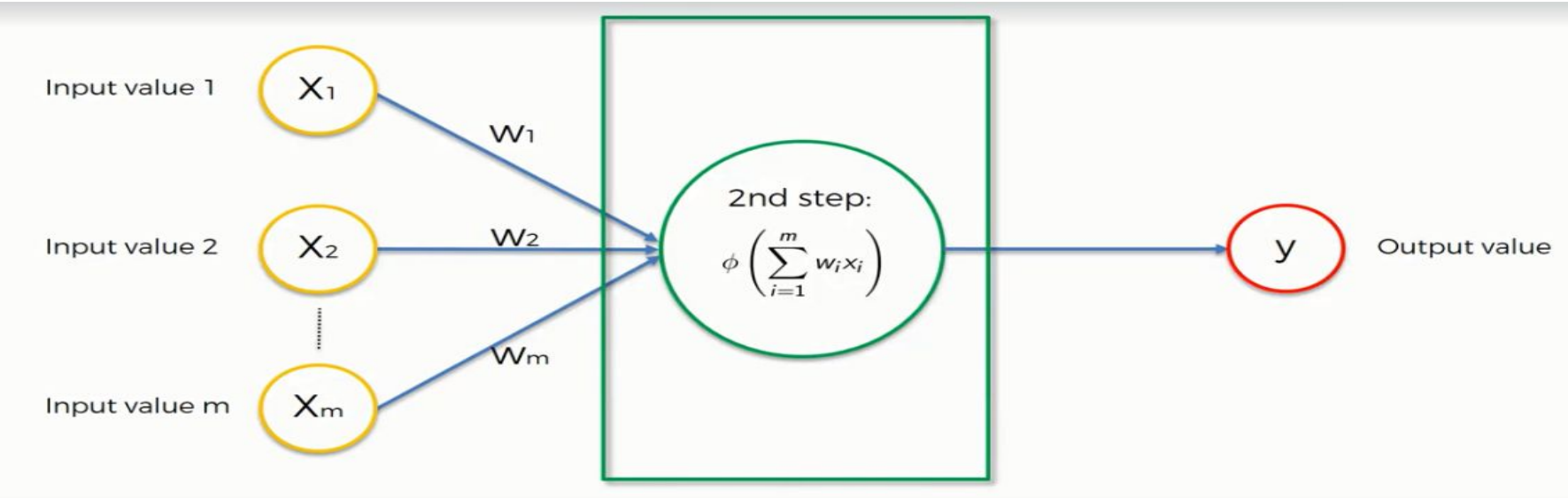
- Input Layer:
  - It represents one row in the observation.The values of the observation are standardized or normalized.This has to be normalized since all values should be in the same range.
- Synapse:
  - Synapses are connections between neurons through which "information" flows from one neuron to another.
- Output:
  - The output can be continuous, binary and categorical output.If in case of categorical values, we obtain multiple values as output.
- Working:
  - Neural networks learn by adjusting weights.While the inputs pass through the synapses weights are assigned to the input. Later gradient descent and backpropagation are used for adjusting the weights.

Steps of neural network working - 1



Step 1: All the inputs and weights are added up.

Steps of neural network working - 2



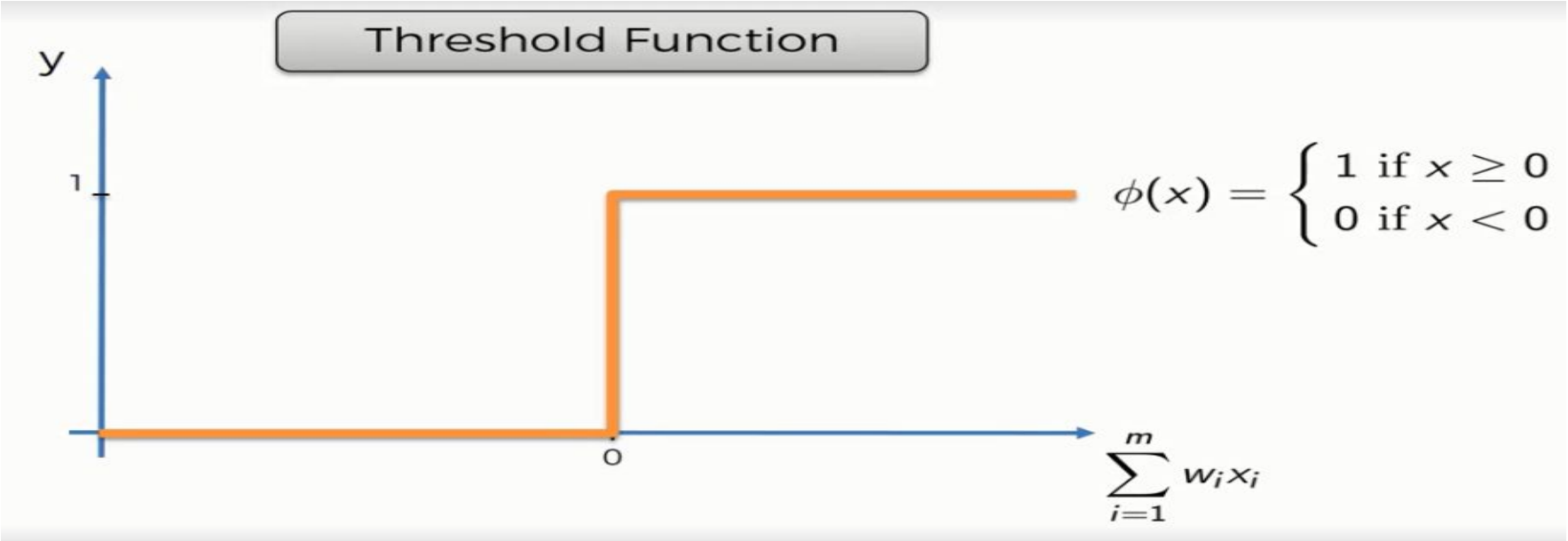
Step 2 : Then activation function is applied to weighted sum signal. Based on this neuron will decide whether or not to pass the signal.These steps are repeated again and again thousands of times depending upon the number of neurons.

Activation Function

There are many functions which are used as activation functions.Four of the most predominant functions are:

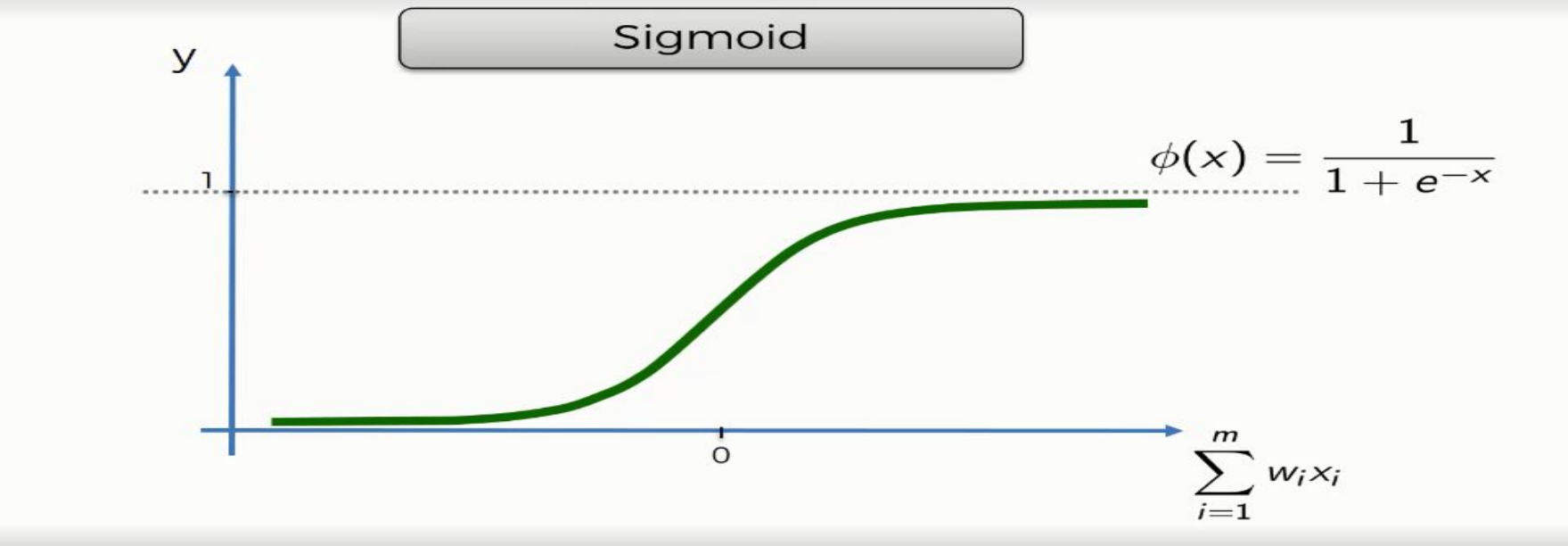
- Threshold Function
- Sigmoid Function
- Rectifier Function
- Hyperbolic Tangent

Threshold Function



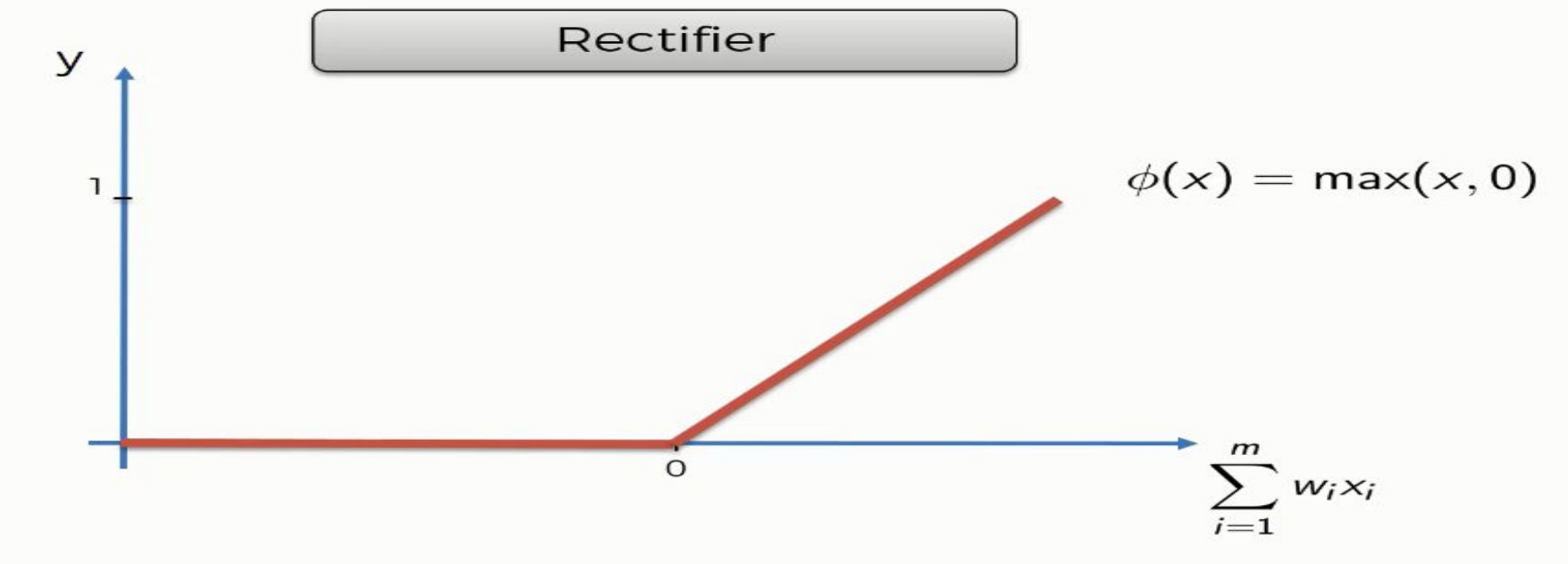
If the value is less than zero then the threshold function passes on 0 else it passes on 1.

Sigmoid Function



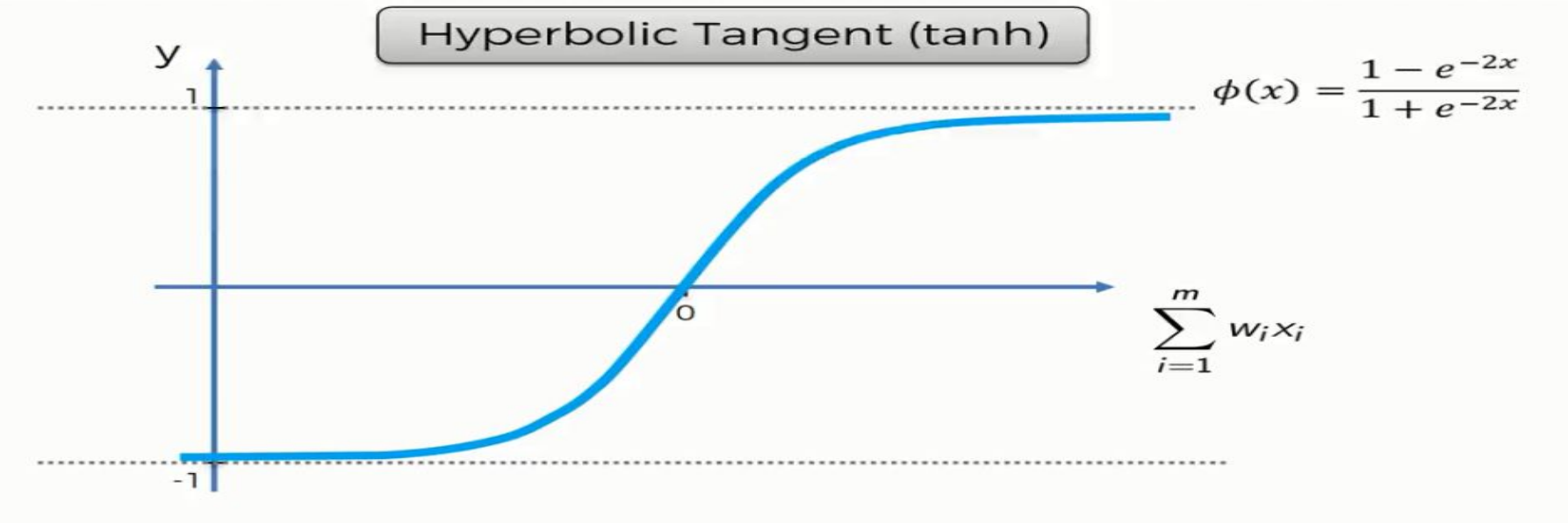
This is the function used in logistic regression. If the values of the function is less than zero then the output value is taken as zero else it is taken as 1. It is used for probabilistic prediction.

Rectifier Function



This function progresses to zero and then gradually increases to 1.It is the mostly used function in neural network.

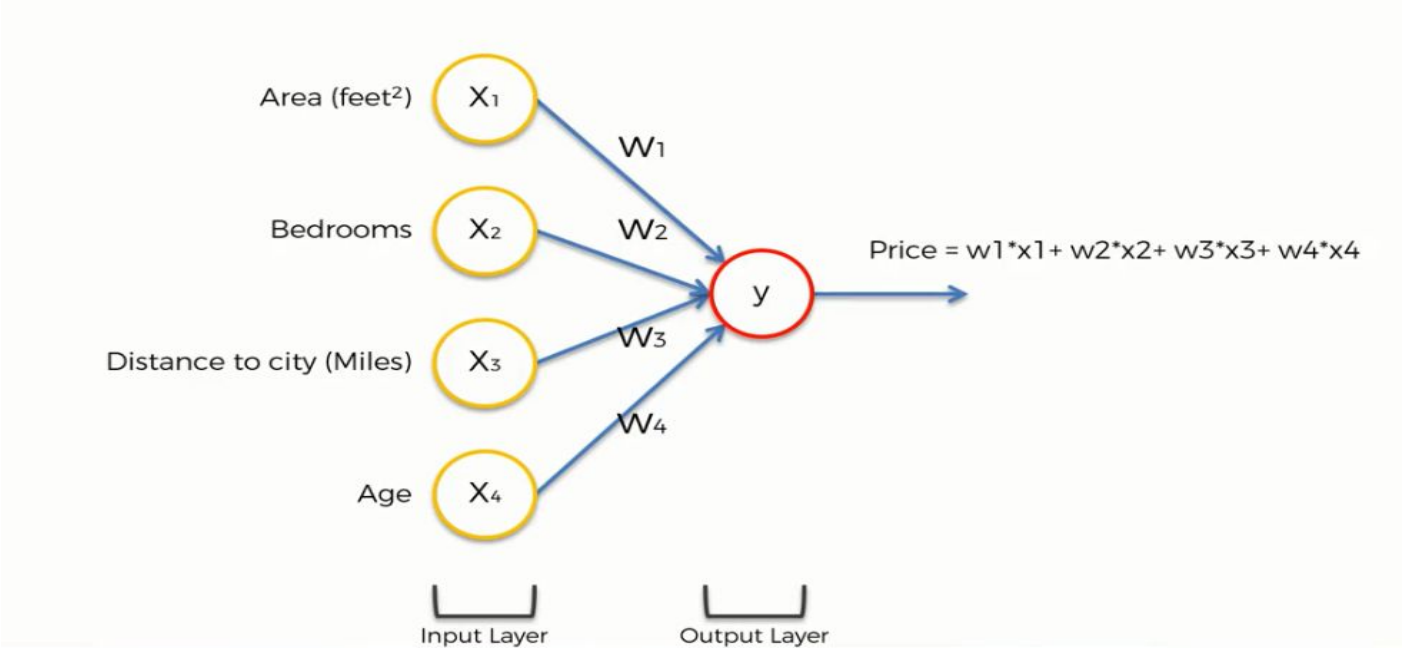
Hyperbolic Tangent



It is similar to sigmoid function but values go from 0 to 1 and then from 0 to -1

Example of working of ANN:

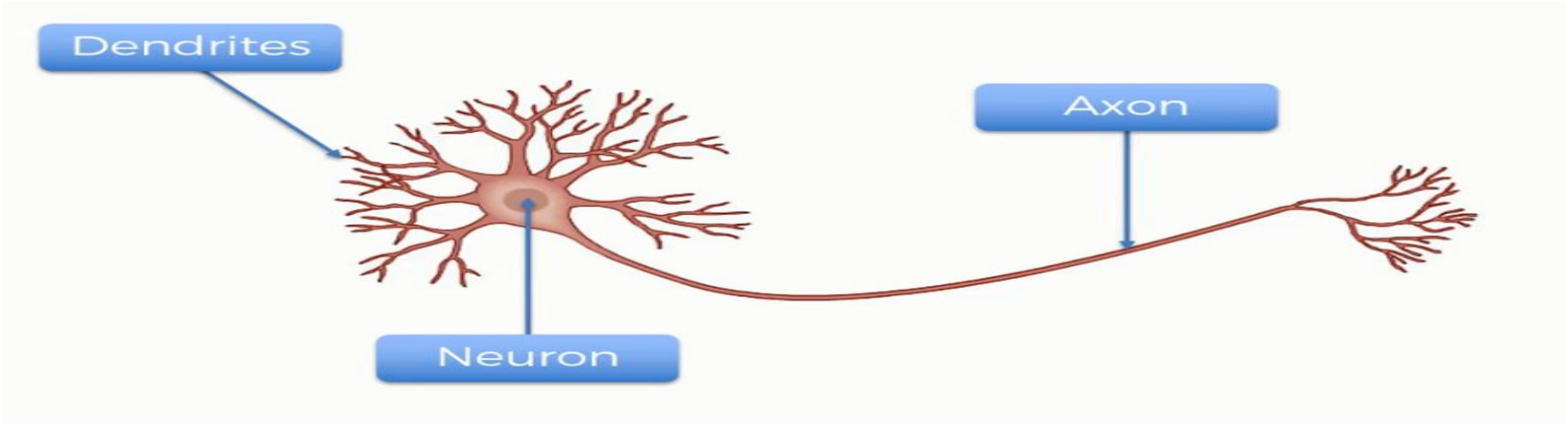
If we consider price prediction of a flat based on the features of the house like no.of bedrooms, Age of the house etc.The general representation of the neural network is:



Example of working of ANN:

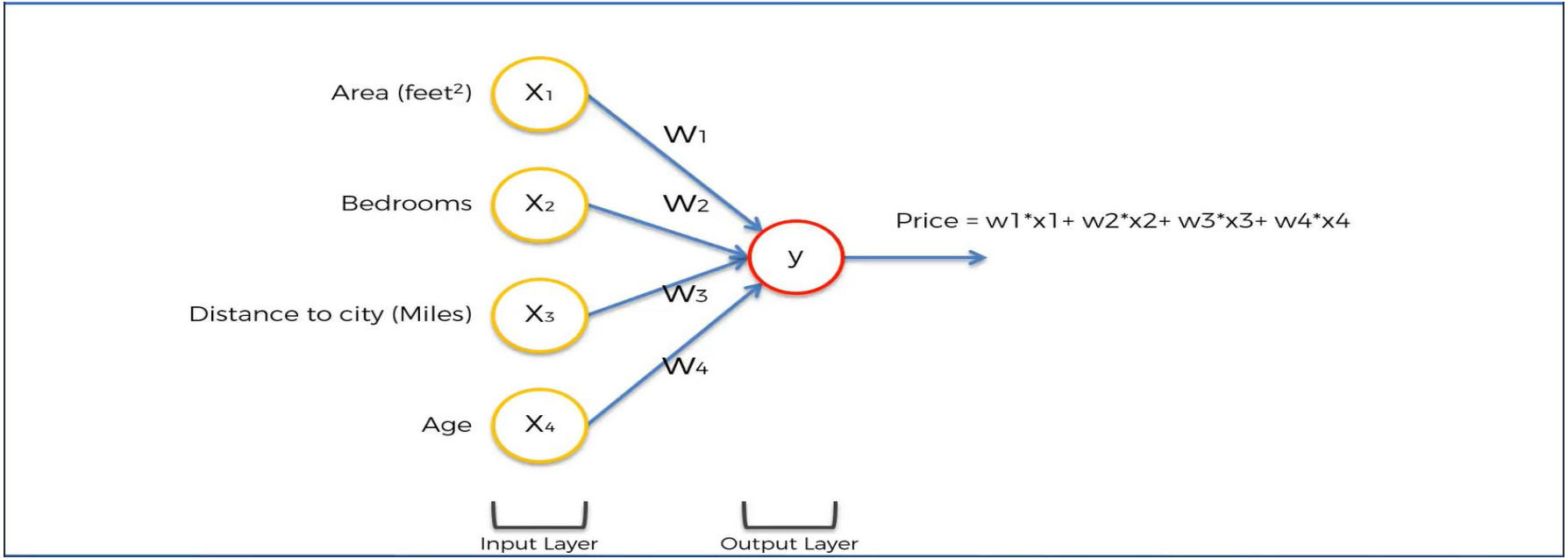
- The representation shown in the previous slide is a basic representation but the power of neural networks comes from the concept of using hidden layers.
- All the features are not connected to all the neurons. The connection of features depends on the specific function which is done by the neuron.
- Ex: Suppose if the function at a neuron is looking for houses which are not too far away and spacious then only those features are connected to it.

Neuron



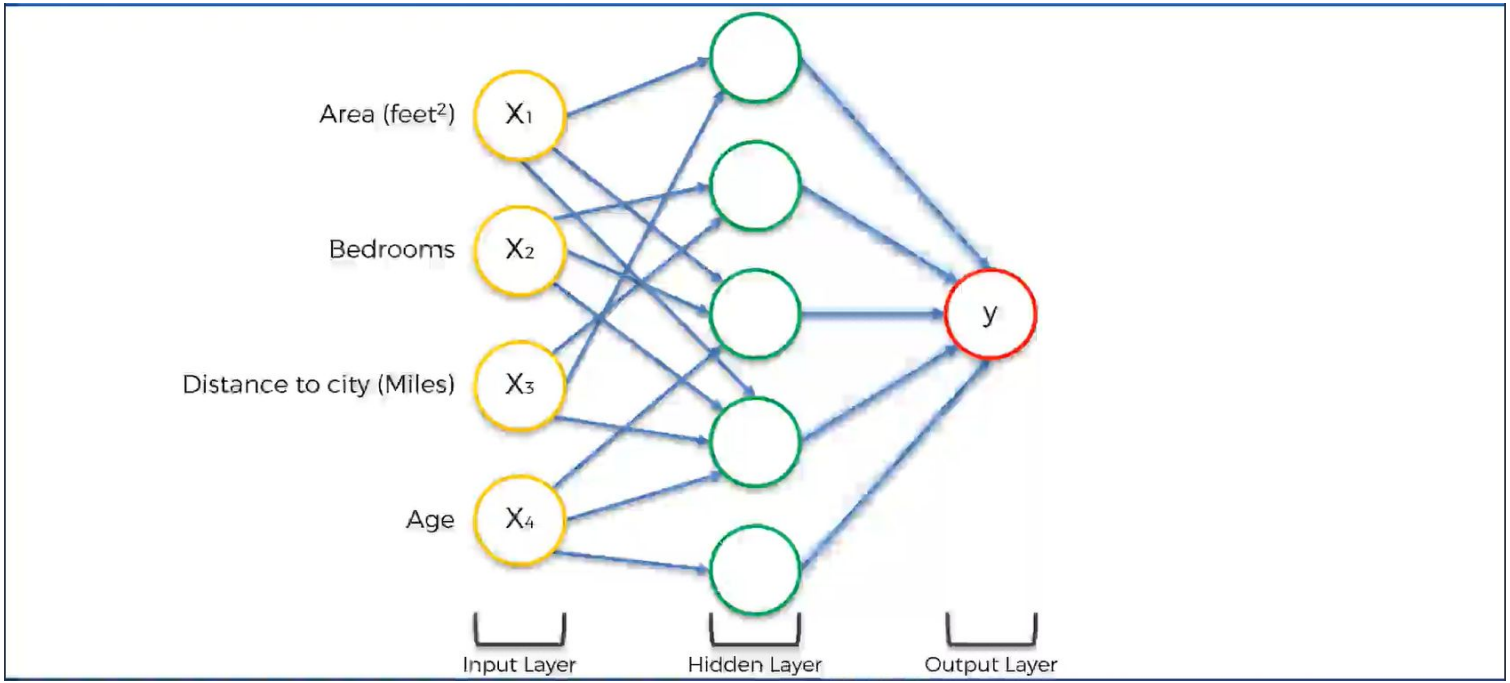
- Neurons are basic building blocks of artificial neural networks.
- Dendrites and Axons help neurons to work together.Dendrites are receivers of signal and Axons are transmitters of signals.

ANN with a simple activation function



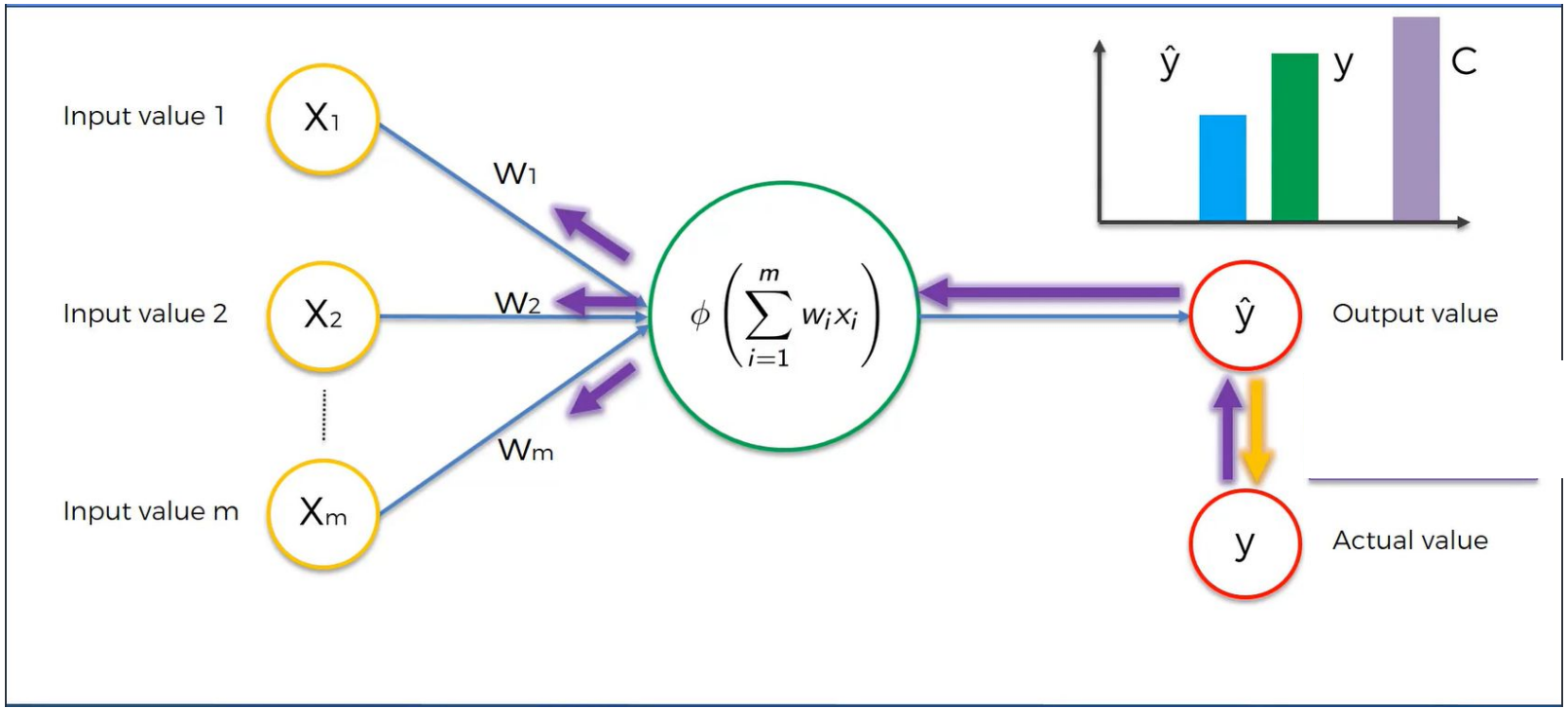
- Price of the house is calculated using simple Linear Regression that is used as Activation Function in this ANN.

ANN with hidden layer



- Each node in the hidden layer searches for a set of features.

Cost function and weights updation



ANN - Seven Step Process

STEP 1: Randomly initialise the weights to small numbers close to 0 (but not 0).

STEP 2: Input the first observation of your dataset in the input layer, each feature in one input node.

STEP 3: Forward-Propagation: from left to right, the neurons are activated in a way that the impact of each neuron's activation is limited by the weights. Propagate the activations until getting the predicted result y.

STEP 4: Compare the predicted result to the actual result. Measure the generated error.

STEP 5: Back-Propagation: from right to left, the error is back-propagated. Update the weights according to how much they are responsible for the error. The learning rate decides by how much we update the weights.

STEP 6: Repeat Steps 1 to 5 and update the weights after each observation (Reinforcement Learning). Or: Repeat Steps 1 to 5 but update the weights only after a batch of observations (Batch Learning).

STEP 7: When the whole training set passed through the ANN, that makes an epoch. Redo more epochs.

Cost function and weights updation

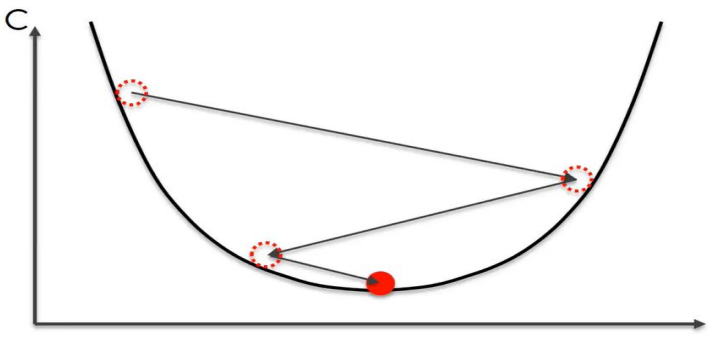
- Input is fed into the Neural Network and the predicted output is obtained using the Activation function.
- The predicted output is compared to the actual output using the cost function.
- Cost Function:

$$C = \frac{1}{2}(\hat{y} - y)^2$$

- To get the minimized cost function the weights are varied.
- The best minimized cost function will give the most accurate results.

Gradient Descent

- Used to get the best weights for all the Synapses.
- Gradient Descent searches for the local minima for the cost function.
- But using Gradient Descent will take many years to complete the process of getting the best weights.



$$C = \frac{1}{2}(\hat{y} - y)^2$$

Stochastic Gradient Descent

- The Gradient Descent method may miss the global minima if the graph is not a convex shaped one.
- To avoid that Stochastic Gradient Descent is used.
- In Gradient Descent all the rows are read and used to generate the cost function.
- In Stochastic Gradient Descent, each row is taken and fed into the neural network.
- The weights are then updated and this process continues for every row in the input data.

Back Propagation

- In forward propagation, the assumed weighted are used to build the ANN.
- The errors are calculated using the Cost Function, and these errors are back propagated to adjust the weights.
- Instead of checking the involvement of the weights in the error and adjusting them individually, the Back Propagation mechanism helps in adjusting the involvements of all the weights at one shot. (Example?)

Back Propagation